Knowledge Acquisition by Demonstration

Henry Lieberman

Media Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
lieber@media.mit.edu

Abstract

Traditional knowledge acquisition tools rely either on conversational textual interfaces or on graphical editing of rule and description structures. Both these methods divorce the user from the actual concrete problem-solving situations where the knowledge is created and applied. We show how a programming by demonstration system can be used as a knowledge acquisition tool by allowing a user to demonstrate procedures interactively on images taken from a video of the procedure being performed in the real world. Graphical annotation and speech input and output are used to provide declarative information that informs the system about objects in the scene, and how to interpret the user's actions.

Introduction

What is the relation between the seemingly disparate research areas of Knowledge Acquisition, Machine Learning, and Programming by Demonstration? All are concerned with getting human knowledge into the machine in one way or another, but approach the problem from different angles. Knowledge Acquisition is motivated by the concerns of traditional expert systems, and promotes conversational interactions to elicit knowledge in the form of rules that drive expert system inference engines. Machine Learning deals with the problem of abstraction of functions from concrete data, and Programming by Demonstration is concerned with learning procedures by generalizing recorded user actions.

Knowledge acquisition has traditionally placed too much emphasis on the role of intermediaries: a knowledge engineer who interviews the domain expert and constructs a knowledge-based system, and a textual intermediate langauge [such as a rule-based language] in which the knowledge is expressed. Future knowledge acquisition systems should place emphasis on building interactive systems which can be used directly by domain experts, and interfaces which allow the experts direct interaction with media artifacts such as images, video, and sound that document specific examples from which knowledge is derived. Rules or textual language descriptions should be generated automatically by the system as a by-product of such interaction.

Machine learning has traditionally placed too much emphasis on the abstract problem of inducing a functional relation between data representing specific examples and generalized descriptions (Lieberman, 1995). Techniques for solving such problems are important, but it is often difficult to make decisions about which characteristics of algorithms are important [such as the proper *bias* for the learning algorithm] when the problems are divorced from user interaction and the context of specific examples. Techniques that rely on emergent statistical properties of large numbers of examples [e.g. genetic algorithms and connectionist techniques] are often inapplicable when there is a one-to-one correspondence bewteen interactive user actions and the examples. Machine learning should reorient itself toward learning methods that learn quickly, provide good interactive feedback and explanation, work incrementally, and take advantage of explicitly presented user advice.

Programming by demonstration has traditionally been concerned with acquiring procedural knowledge by recording user actions while working on concrete, visually presented examples, in the context of an interactive system such as a text editor or graphical editor. The user presumably applies the domain-specific knowledge in the particular example, and the system acquires knowledge by generalizing the example, perhaps guided by hints explicitly presented by the user in the interface. The challenge for Programming by Demonstration is to move from purely procedural knowledge to include more declarative knowledge, and give the user better means for controlling the generalization process.

Traditional interactive knowledge acquisition tools

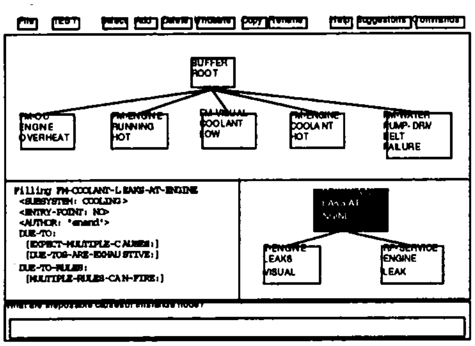
These tools typically divide themselves into two categories. The first adopts a conversational natural language interface, and tries to automatically generate some of the interaction between the knowledge engineer and the expert (Marcus 1988).

```
Indicate whether OXYGEN_READING
>> HIGH NORMAL LOW [NORMAL] <cr>
Indicate whether SMALL-RED-FLAME
>> YES NO [NO] <cr>
LOSS-IN-GAS is explained by HIGH-MOISTURE-
CONTENT or LARGE-PARTICLES.
Do you want a more detailed account?
>> [YES] <cr>
LOSS-IN-GAS is explained as follows:
```

A conversational interface

There are also some tools for managing sets of transcribed interviews and protocol analysis, which also deal only with text.

The second kind provides a graphic tool for visualizing the knowledge structures constructed by the knowledge engineer. Most projects of this sort are box-and-arrow browsers and editors for the knowledge structures produced in a conventional knowledge engineering methodology. The following example is redrawn from (Marcus 1988).



A knowledge structure editor

However, nowhere do concrete visual representations of examples play a significant role. The examples are all described verbally by the expert or by the knowledge engineer. No direct connection between the concrete visual examples and the knowledge structures is provided.

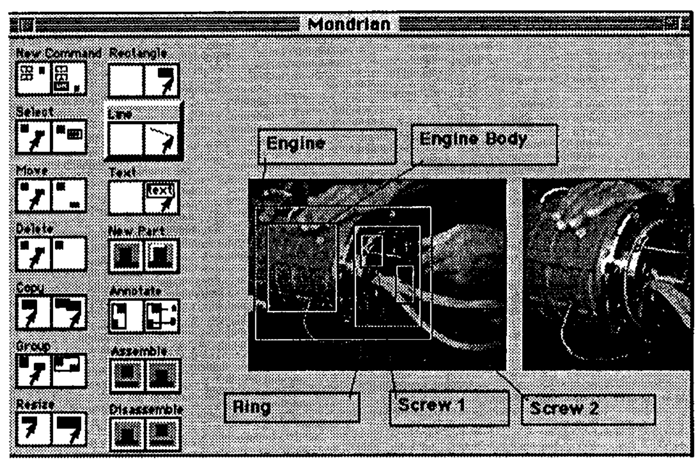
Mondrian: Teaching procedures from video

Mondrian (Lieberman 1993) is a Programming by Demonstration graphical editor that has been extended with facilities for knowledge acquistion. Mondrian is set up to learn procedures such as assembly and maintainance procedures for a variety of machines. Rather than adopt the traditional knowledge engineering methodology of having a knowledge engineer interview an expert technician, Mondrian is designed to be used by the technician directly, who may not have any knowledge of AI or programming.

Much expert knowledge is strongly *situated*. Expert knowledge arises out of concrete situations, and experts are best at deciding what knowledge to apply and how to use it when they are placed in real situations. Furthermore, experts are best at articulating their knowledge, as when they are trying to teach an apprentice, when they can explain their knowledge in a concrete and realistic situation.

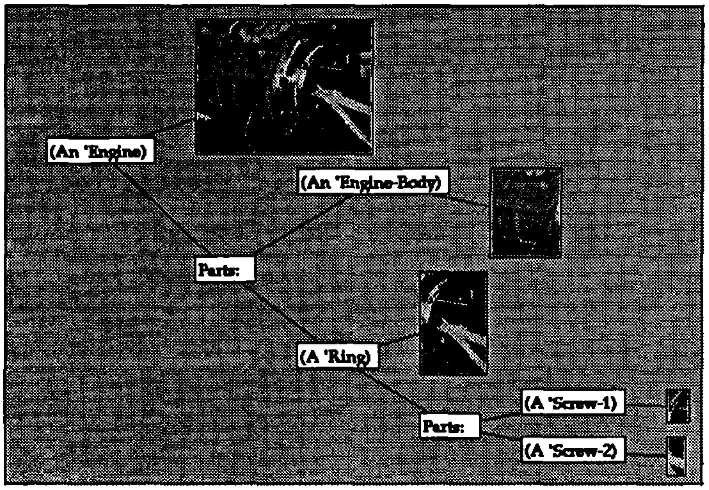
In order to take advantage of this, we try to elicit the knowledge by using interactive media to simulate a concrete situation. We do this by using a digitized video of the procedure to recreate a concrete example situation. The expert then explains the procedure to the machine by a combination of procedural and declarative techniques.

The procedure to be communicated consists of a sequence of steps. For each step, the expert selects a pair of video frames that represent before- and after- frames. Each step is demonstrated by applying interactive operations to the before frame with the goal of achieving the final state represented by the after frame. The interactive steps are recorded by Mondrian, using the standard methodology of Programming by Demonstration.



However, most Programming by Demonstration systems are weak in communicating new declarative knowledge. New declarative knowledge is necessary, because we must teach the system about the objects and object relations in the video. Our system does not rely at all on computer vision or recognition to understand the objects in the video frame. The location of each object is explicitly drawn on the video frame by the user.

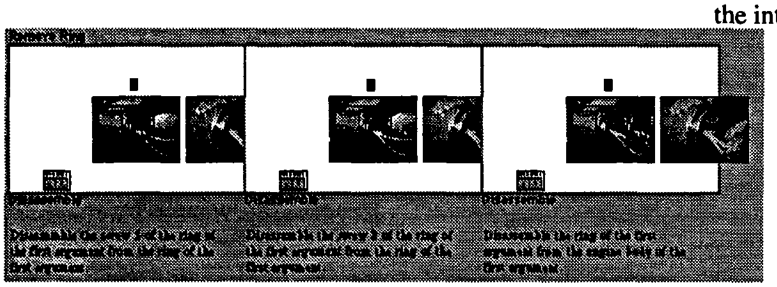
The user specifies parts by drawing textual labels and lines from the labels to the parts they describe. The selected subsets of the video frame provide a visual rerepresentation of the objects and their part-whole relations. The system then displays a graph of the conceptual structure, linked to the images of the individual parts in each example. Annotation structures can be copied and use to annotate subsequent frames. The system has provision for identifying objects that remain the "same" across frames, although they are represented by different images.



This representation has the advantage of integrating the conceptual structure represented by the tree of parts, with the concrete images in a particular situation.

The part images represent a virtual environment for demonstrating the steps of a procedure. Interactive operations performed on the part images represent operations on the objects they represent. The steps are recorded and generalized to form a procedure, using an incremental form of explanation-based generalization.

Feedback on what is being learned is paramount for user acceptance. We represent the procedure in the form of a storyboard, which integrates saved graphic pictures of each step with a natural language generator that reads an English description of the recorded procedural code.



The system also integrates speech output, speaking the description of each step after it is performed, so that the user understands what the system's interpretation of his or her actions is.

Speech input allows the user to give verbal advice to control Mondrian's generalization (Stoehr and Lieberman 1995). When the user is performing an action, the user may speak any one of a set of [predefined] relations that intercept the generalization process and choose a desired generalization. Speech input is ideal for this purpose because it does not interfere with the hand-eye coordination of the graphical manipulation task. It gives a natural "show and tell" feeling to the interface.

Experience with Mondrian

The MIT Media Lab is collaborating with the Italian electronics company Alenia on applications in the technical documentation field. Alenia makes many specialized electronic devices for the aerospace industry, such as air traffic control radar systems. Like many companies, a large number of the devices produced by Alenia are characterized by the following attributes: They are complex devices, have relatively small user communities, must be operated by personnel untrained in the design of the devices, operated under time constraints, and the consequences of operational mistakes or equipment failure can be expensive or dangerous. All these factors conspire to make production of documentation for operational and maintenance procedures for the devices expensive, time-consuming and error-prone.

We are exploring the use of Mondrian to aid in producing technical documentation for electronic and mechanical devices. A relatively untrained technician who is familiar with the operation of the device can "teach" the computer the steps of the procedure by interacting with an on-screen simulation of the device. Such a procedure could also be used to drive robotic assembly or test equipment on an assembly line. The automatially-produced documentation may not be equal in quality to hand-written professional documentation, but it faithfully and inexpensively records the interaction, guarding against such common problems as omitted or misrecorded steps.

Far from being a "frill", we found the multimodal character of Mondrian's interface to be essential for its success in integrating knowledge acquisition with the programming by demonstration framework. The use of video of procedures performed in the real world by actual experts, graphical annotation of real images, graphical and speech output by the system to communicate its interpretations of the user's actions, and speech input to give advice to the sytem about generalization all contribute to the sense of realism that makes it work.

Acknowledgements

This work was sponsored in part by Alenia, Apple Computer, ARPA/JNIDS and the National Science Foundation., and other sponsors of the MIT Media Laboratory.

References

Boose J.H. 1989. A Survey of knowledge acquisition techniques and tools. *Knowledge Acquisition* **1**, pp. 3-37.

Lieberman, H. 1993. Mondrian: A Teachable Graphical Editor, in Allen Cypher, ed. *Watch What I Do: Programming by Demonstration*, MIT Press, Cambridge, MA.

Lieberman, H. 1994. A User Interface for Knowledge Acquisition from Video, in Conference of the American Association for Artificial Intelligence, Seattle, August 1994.

Lieberman, H. 1995. Interaction Is The Key To Machine Learning Applications, Workshop on Learning from Examples and Programming by Demonstration, International Conference on Machine Learning, Lake Tahoe, California, July 1995.

Marcus, Sandra, ed. 1988. *Automating Knowledge Acquisition for Expert Systems*, Kluwer Academic Publishers, 1988.

Stoehr, E. and Lieberman, H. 1995. Hearing Aid: Adding Verbal Hints to a Learning Interface, ACM Multimedia Conference, San Francisco, November 1995.